THE EMIRATES

2020-2021

Creative Design & Innovation

Teacher Guide

Grade 11

Creative Design and Innovation

G11 Advance Teacher's Guide

CREATIVE DESIGN
INNOVATION

Term 2

2020-2021

Contents

Introduction:

This Teacher's Guide aims to provide the teachers of Creative Design and Innovation with a set of teacher support materials. This includes the Instructional Planner (IP), the Lesson Plans (LPs) and Answer Keys.

The Instructional Planner aims to provide teachers with the scope and sequence during the term. Teachers will be able to have a detailed idea of when to teach each section of the book and accordingly organise their work during the entire term in advance. This IP can be found on LMS.

The Lesson Plans provide a model teaching strategy for Creative Design and Innovation teachers. It highlights the core points that allows teachers to support the progress of their students. It also provides a plenty of advices for the teachers to follow in class promoting various teaching methodologies, practices and strategies. It contains answer keys for all the questions and activities within the book, in order to provide teachers with model answers that guarantee a moderate and consistent level for answers across the country.

As a CDI teacher for Grade 11 students, you have a great responsibility of enlightening your students with the available opportunities in their higher education studies. CDI is a very important and rich subject that eventually feeds into many science engineering and design fields. Please demonstrate this importance at the beginning and throughout the CDI course. This will allow students to give extra attention to the subject and motivate students to explore the subject outside the classroom.

Wishing you a very successful and fruitful term with your creative and innovative students!

The authors,

December 2020.

Instructional Planner:

Trimester Planner
Term two (T2) 2020/2021

SUBJECT: Creative Design and Innovation (CDI)

Grade **11**
Advance

Overview:

This Instructional Planner contains a traditional **Weekly Planner** which outlines the anticipated Scope and Sequence for teaching the course during the term, detailing SLOs per week. The IP also highlights what resources are needed to complete the relevant sections of the workbook throughout the weeks.

The CDI curriculum features **two main books** for students to use. The structure of the three books is outlined below:

- **Book #1**: **Student's Book**:
  - This is students' reference book and their main learning resource.
  - SB will be available **online ONLY** for students and teachers.

- **Book #2: Workbook:**
  - This is students' resource for documenting their work. **It is accordingly the main book for students to use**. Completing the workbook means completing the course and achieving all outcomes required.
  - WB will be **printed for each student**.
  - WB will also be **available online** for students.

- **CDI Books Printing Matrix:**

Book	Printed per Student	Online
Student's Book	×	√
Workbook	√	√

Grade **11**
Advance
Weekly Planner (SLOs breakdown)

Note:

- All learning outcomes are essential unless highlighted in Green.
- The learning outcomes in **Bold** are from the Stream project.

Week	Chapter	Section	Books	Overview	Learning Outcomes	WB Activities	DL Platforms	Resources	Can be covered through		
									Distance learning	Self-learning	Self-study
1 & 2	4	1	SB/ WB	**Section 1:** Introduction to software lifecycle	• Define the software development lifecycle.	4.1.1 – 4.1.8	Diwan	Laptop	✓	✓	
					• Describe a software development lifecycle.		Diwan	Laptop	✓	✓	
					• Distinguish between various software development models.		Diwan	Laptop	✓	✓	
					• Apply different requirement analysis tools.		Diwan	Laptop	✓	✓	
					• Describe abstraction using real-life examples.		Diwan	Laptop	✓	✓	
3 & 4	4	2	SB/ WB	**Section 2:** Software testing	• Assess the importance of testing software.	4.2.1 – 4.2.5	Diwan	Laptop	✓	✓	
					• Create a test plan.		Diwan	Laptop	✓	✓	

5	5	1	SB/ WB	**Section 1:** Introduction to OS	• Illustrate the purpose of an operating system.	5.1.1 – 5.1.2		Laptop	✓	✓	
					• Compare the different types of software.			Laptop	✓	✓	
6	6	1	SB/ WB	**Section 1:** Basic Principles of IoT	• Demonstrate the basic principles of IoT.	6.1.1 – 6.1.2		Electronic Fundamentals Kit Oryx kit Laptop	✓	✓	
					• Examine the main structure and components of an IoT system.			Electronic Fundamentals Kit Oryx kit Laptop	✓	✓	
					• Use an IoT platform to manipulate a controller board.			Electronic Fundamentals Kit Oryx kit Laptop	✓	✓	
7	6	2	SB/ WB	**Section 2:** Control system inputs	• Construct a system using digital and analogue sensors.	6.2.1 – 6.2.6		Electronic Fundamentals Kit Oryx kit Laptop	✓	✓	
					• Develop a computer program to control various electronic components.			Electronic Fundamentals Kit Oryx kit Laptop	✓	✓	
					• Explain the function of sensors in electronic circuitry.			Electronic Fundamentals Kit Oryx kit Laptop	✓	✓	
					• Recommend various electronic sensors for suitable applications.			Electronic Fundamentals Kit Oryx kit Laptop	✓	✓	

7	6	3	SB/ WB	**Section 3:** Using Microprocessor Boards	• Control a microprocessor-based system.	6.3.1 – 6.3.4		Electronic Fundamentals Kit Oryx kit Laptop	✓	✓	
					• Evaluate and modify codes to best solve a problem.			Electronic Fundamentals Kit Oryx kit Laptop	✓	✓	
8	WB: Stream project	1	WB /SB	**1. The design brief**	• **Analyse the main sections of a design brief.** • **Address the constraints and requirements of a design problem.**	-		Laptop	✓	✓	
				2. Analysing the brief	• **Analyse the main sections of a design brief.** • **Address the constraints and requirements of a design problem.**	1.2.1 – 1.2.3		Laptop	✓	✓	
9	WB: Stream project	1	WB /SB	**3. Research and Investigation**	• **Apply different methods of research.**	1.3.1 – 1.3.2		Laptop	✓	✓	
				4. Possible solutions	• **Transform research ideas into possible solutions for a design problem.**	1.4.1 – 1.4.8		Electronic Fundamentals Kit Oryx kit Laptop	✓	✓	
				5. Final Programme	• **Construct a prototype to solve a design problem through various prototyping means.** • **Test a design prototype for operational expectations.**	1.5.1		Oryx kit Laptop	✓	✓	

10	WB: Stream project	1	WB /SB	**6. Design realisation**	• **Construct a prototype to solve a design problem through various prototyping means.** • **Test a design prototype for operational expectations.** • **Improve a design prototype to overcome identified design faults.**	1.6.1 – 1.6.2		Electronic Fundamentals Kit Oryx kit Laptop	✓	✓	
				7. Evaluation	• **Evaluate the success of implementing the proposed design idea.** • **Improve a design prototype to overcome identified design faults.** • **Evaluate the success of implementing the proposed design idea.**	1.7.1		Electronic Fundamentals Kit Oryx kit Laptop	✓	✓	

LESSON PLAN

Grade: 11 A

Chapter 4: Software Lifecycle
Section 1: Introduction to software lifecycle

1) Learning outcomes

Essential LOs

- ☐ Define the software development lifecycle.
- ☐ Describe a software development lifecycle.
- ☐ Distinguish between various software development models.
- ☐ Apply different requirement analysis tools.
- ☐ Describe abstraction using real-life examples.

Strategies/Activities

Distance learning

- ★ LMS discussion forms
- ★ Microsoft teams group meetings, chats and assignments
- ★ E-Surveys
- ★ E-Polls

2) Keywords

- Software
- Software lifecycle
- Server

3) Resources

- Computer
- Student Book
- Workbook
- LMS
- Microsoft teams
- Tinkercad
- Video tutorials

4) Prior knowledge

- Computer programming

5) Assessment

Assessment for learning

- ★ Observations
- ★ Conversations
- ★ Notes
- ★ Work sample
- ★ Checklist
- ★ Diagnostics

Assessment as learning

- ★ Self-assessment
- ★ Peer-assessment
- ★ Presentation
- ★ Graphic Organizer
- ★ Collaboration
- ★ Homework

Assessment of learning

- ★ Workbook activities
- ★ Test
- ★ Quiz

6) Starter

Distance learning

- At the beginning of the lesson, introduce students to the lesson aim.
- Teacher to ensure all learning outcomes are outlined on LMS under the correct lesson.
- Use Microsoft teams screen sharing to show all learning outcomes and ensure understanding.
- Introduce the lesson's keywords through an LMS quiz or through an online game-based learning platform such as Kahoot or Quizlet.

7) Differentiation strategies

Differentiation to be met through designing various levels of activities. Some examples are outlined below:

Distance learning

- ★ LMS quizzes with visual aids MCQ, true or false and multi – response questions.
- ★ Differentiated LMS in class activities assigned to specific students.
- ★ Reflection through LMS surveys or polls.

Identify possible differentiation needs of the class based on prior classes and the starter activities.

8) Lesson activities

Distance learning

Teacher led activities	Student led activities
☐ Teacher to start an online lesson on Microsoft teams. ☐ Teacher to upload reading materials to LMS prior to the lesson. ☐ Students can view the books online via Al Diwan platform. ☐ Teacher to lead the class discussion on what is meant by software and ask students to give examples on them. ☐ Teachers to lead a class discussion on the types of software. ☐ Teachers to make sure that students complete activity 4.1.1. ☐ LMS activity ☐ Teachers to introduce the seven SLDC phases and discuss with students what is required in the planning phase. ☐ Teachers to make sure that students complete activity 4.1.2. ☐ LMS activity ☐ Teachers to explain the requirement analysis phase, highlighting the difference between functional and non-functional requirements. ☐ Teachers to make sure that students complete activity 4.1.3. ☐ LMS activity ☐ Teachers to explain the design and prototyping phase. ☐ Teachers to make sure that students complete activity 4.1.4. ☐ LMS activity ☐ Teachers to explain the software development phase. ☐ Teachers to make sure that students complete activity 4.1.5. ☐ LMS activity ☐ Teachers to explain the testing phase and discuss with students the importance of this phase when developing any software. ☐ Teachers to make sure that students complete activity 4.1.6. ☐ LMS activity ☐ Teachers to explain the deployment and the operations and maintenance phases. ☐ Teachers to make sure that students complete activity 4.1.7. ☐ LMS activity ☐ Teachers to explain the models of SDLC phase.	☐ Students to log on to Microsoft teams lesson ☐ Download any required material for the lesson from LMS. ☐ Open the required book on Al Diwan. ☐ LMS activity Students to complete activity 4.1.1. ☐ LMS activity Students to complete activity 4.1.2. ☐ LMS activity Students to complete activity 4.1.3. ☐ LMS activity Students to complete activity 4.1.4. ☐ LMS activity Students to complete activity 4.1.5. ☐ LMS activity Students to complete activity 4.1.6. ☐ LMS activity Students to complete activity 4.1.7.

☐ Teachers to make sure that students complete activity 4.1.8. ☐ LMS activity	
Workbook activities ★ **4.1.1 – 4.1.8**	**9) Cross-curricular links**  ▪ (Teacher to make cross-curricular links and activities where possible.)

10) Plenary

- Teacher to facilitate as students evaluate learning.
- Question pupils on what they have learned. Have learning outcomes been met? Has the lesson aim been achieved?
- Teacher to use LMS quiz or an online game-based learning platform such as Kahoot or Quizlet to check students understanding of the lesson objectives.
- Teacher to post homework activities through LMS and to assign a time frame for that for all unfinished activities.
- Teacher to create an LMS/Google/Microsoft surveys and share it with students where a checklist showing students' reflection will help self-reflect on the lesson for both teacher and students.
- Teacher to use LMS awards to give students recognition badges.

11) Reflection & Next steps

Activities that worked	Topics to be revisited

Answer Key – Workbook

Activity 4.1.1

Which operating system (OS) is used to run your laptop?

- Microsoft windows

Name two types of application software you use daily.

- Answers may vary

Activity 4.1.2

For the following projects, recommend a suitable business analyst to join your team.
Hint: Analysts must have a good knowledge of the business domain.

Business	**Analyst**
Healthcare project	A doctor or someone who has healthcare knowledge
Financial project	Charted accounting or someone with an MBA in finance

Activity 4.1.3

Identify the functional and non-functional requirements for the following products. *The first one is done for you.*

Product	**Functional requirement**	**Non-functional requirement**
Cup	ability to contain tea or coffee without leaking	contain hot liquid without heating up to more than 45°C
Milk cartoon	ability to contain fluid without leaking	ability to contain a gallon of milk
Construction hat	must not break under pressure of less than 10,000 PSI	must be comfortable to wear

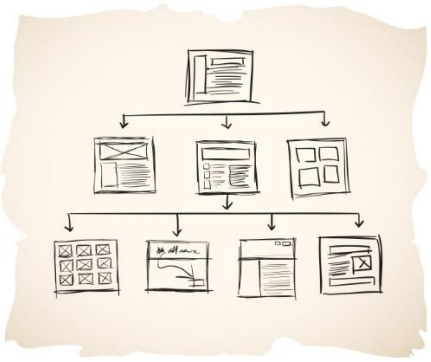
Now, identify the Functional and non-functional requirements for the following software products.

Product	**Functional requirement**	**Non-functional requirement**
Facebook	profile photo on Facebook should be visible on login posts can be 63,206 characters long	time to log in (performance) a post will appear to other users' lists within 30 seconds
Microsoft teams	run on computers with memory 4.0 GB RAM	The software should be portable, available on different OSs

Difference: non-functional requirements describe how the system works, while functional requirements describe what the system should do.

Activity 4.1.4

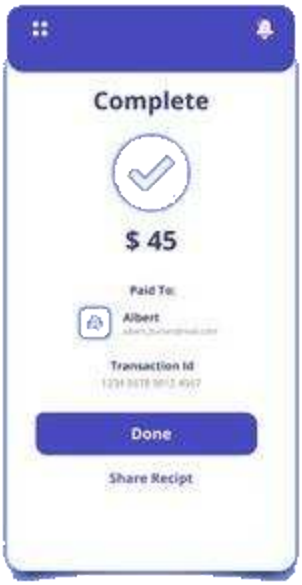
For the banking website, multiple pages/modules are required, where each page is made for different but related functionality. Design three pages of the website and show how these pages interact with each other.
Note: To complete this activity, you may use online web design tools.



Page 1	Page 2	Page 3

Pages interaction:

Answers may vary



Activity 4.1.5

Match the software development jobs on the left with the correct description on the right using arrows.

Developer	Continue planning the user interface.
Operation	Analyse the requirements and start building test cases for their test plans.
Designer	Write the code of the application.
Tester	Set up the physical hardware for the servers.

Activity 4.1.6

Carry out a quick research. List two possible issues or defects that can be identified while testing a software program.

Answers may vary.

- The website performance is too slow
- The login function of the website does not work properly
- The GUI of the website does not display correctly on mobile devices
- The website could not remember the user login session
- Some links don't work

Activity 4.1.7

Match the SDLC phases on the left with the correct description on the right using arrows.

Planning	Write the code for the software.
Requirement analysis	Plan the program languages, database, architecture, user interface etc. which are suitable for the project.
Software design	Release the product into the production environment.
Software development	Gather information about the details and specifications of the desired software from the client.
Testing	Meet with the client, discuss the terms of the agreement, and sign a deal.
Deployment	Make changes to accommodate enhancements requested by the client.
Maintenance	Validate that the software is built as per the requirements given by the client.

Activity 4.1.8

A software development company is working on a new project where the requirements are not clearly defined yet. Therefore, the company is facing difficulties in completing the project. They decide to build a sample application and show it to the client for feedback.

Assume you were the project manager, which software lifecycle model would you select and why?

The Agile model mainly because the vague and unclear requirements. In this model we can have some sprints, each sprint duration from 2 to 3 weeks, and the team will start to take part of the requirements which it is okay to be not clear.

The team can build a prototype of what they understand and show it to the customer and discuss the customer feedback. This will stimulate the new requirements from the feedback as changes or new additional business requirements need to be developed. Throughout the development lifecycle, the team will build the backlog and repeat the same process with each sprint.

LESSON PLAN

Grade: 11 A

Chapter 4: Software Lifecycle
Section 2: Software testing

1) Learning outcomes

Essential LOs

- ☐ Assess the importance of testing software.
- ☐ Create a test plan

Strategies/Activities

Distance learning

- ★ LMS discussion forms
- ★ Microsoft teams group meetings, chats and assignments
- ★ E-Surveys
- ★ E-Polls

2) Keywords

- Test plan

3) Resources

- Computer
- Student Book
- Workbook
- LMS
- Microsoft teams
- Tinkercad
- Video tutorials

4) Prior knowledge

- Computer programming

5) Assessment

Assessment for learning

- ★ Observations
- ★ Conversations
- ★ Notes
- ★ Work sample
- ★ Checklist
- ★ Diagnostics

Assessment as learning

- ★ Self-assessment
- ★ Peer-assessment
- ★ Presentation
- ★ Graphic Organizer
- ★ Collaboration
- ★ Homework

Assessment of learning

- ★ Workbook activities
- ★ Test
- ★ Quiz

6) Starter

Distance learning

- At the beginning of the lesson, introduce students to the lesson aim.
- Teacher to ensure all learning outcomes are outlined on LMS under the correct lesson.
- Use Microsoft teams screen sharing to show all learning outcomes and ensure understanding.
- Introduce the lesson's keywords through an LMS quiz or through an online game-based learning platform such as Kahoot or Quizlet.

7) Differentiation strategies

Differentiation to be met through designing various levels of activities. Some examples are outlined below:

Distance learning

- ★ LMS quizzes with visual aids MCQ, true or false and multi – response questions.
- ★ Differentiated LMS in class activities assigned to specific students.
- ★ Reflection through LMS surveys or polls.

Identify possible differentiation needs of the class based on prior classes and the starter activities.

8) Lesson activities

Distance learning

Teacher led activities	Student led activities
☐ Teacher to start an online lesson on Microsoft teams. ☐ Teacher to upload reading materials to LMS prior to the lesson. ☐ Students can view the books online via Al Diwan platform. ☐ Teacher to lead the class discussion on what is meant by software testing and ask students to give examples on that topic. ☐ Teachers to illustrate the importance of software testing using real life examples. ☐ Teachers to divide students into groups, assign each group a type of testing to study. Groups can share what they understood with the rest of the class. ☐ Teachers to make sure that students complete activity 4.2.1. ☐ LMS activity ☐ Techers to explain the methods and levels of testing with examples. ☐ Teachers to make sure that students complete activity 4.2.2. ☐ LMS activity ☐ Teachers to explain testing documentation. ☐ Teachers to make sure that students complete activities 4.2.3 – 4.2.5. ☐ LMS activities	☐ Students to log on to Microsoft teams lesson ☐ Download any required material for the lesson from LMS. ☐ Open the required book on Al Diwan. ☐ LMS activity Students to complete activity 4.2.1. ☐ LMS activity Students to complete activity 4.2.2. ☐ LMS activity Students to complete activities 4.2.3 – 4.2.5.

Workbook activities	9) Cross-curricular links
★ **4.2.1 – 4.2.5** **SB activities** ★ **4.2.1**	▪ (Teacher to make cross-curricular links and activities where possible.)

10) Plenary

- Teacher to facilitate as students evaluate learning.
- Question pupils on what they have learned. Have learning outcomes been met? Has the lesson aim been achieved?
- Teacher to use LMS quiz or an online game-based learning platform such as Kahoot or Quizlet to check students understanding of the lesson objectives.
- Teacher to post homework activities through LMS and to assign a time frame for that for all unfinished activities.
- Teacher to create an LMS/Google/Microsoft surveys and share it with students where a checklist showing students' reflection will help self-reflect on the lesson for both teacher and students.
- Teacher to use LMS awards to give students recognition badges.

11) Reflection & Next steps

Activities that worked	Topics to be revisited

Answer Key – Workbook

Activity 4.2.1

As you are now aware, there are two types of software testing, manual testing, and automation testing. Carry out a quick online research. Find out what testers should automate as well as when they should automate the testing process.

What to Automate?

- areas at which a user can make transactions such as the login form or registration forms
- areas where a large number of users can access the software at the same time
- GUI items
- connections with databases
- field validations

When to Automate?

- Test automation should be considered when dealing with:
- large and critical projects.
- projects that require testing the same areas frequently.
- requirements not changing frequently.

Activity 4.2.2

Match the functional testing types on the left with the correct description on the right using arrows.

Unit testing	Testing an end-to-end scenario as if a real customer were using the application.
Integration testing	Testing the software by the user or client to determine whether it can be approved or not.
System testing	Testing performed on each module separately.
Regression testing	Testing combined parts of the software or program to validate that they function correctly.
User acceptance testing	Testing to confirm that a recent program or code change has not adversely affected existing features.

Activity 4.2.3

Choose the correct answer.

1. Which of these is NOT part of a software test plan?
   a. Objectives
   b. Strategy
   c. Risks
   d. Mission

2. What does the objectives section in a software test plan do?
   a. The objectives section states the goals of the plan
   b. The objectives section states the scope of the plan
   c. The objectives section summarises the plan
   d. The objectives section outlines the strategy of the plan

3. What does the scope section do in a software test plan?
   a. The scope section states the goals of the plan
   b. The scope section limits the expectations of the plan
   c. The scope section summarises the plan
   d. The scope section outlines the strategy of the plan

Activity 4.2.4

Choose the correct answer.

1. Non-functional testing is carried out to check:
   a) performance.
   b) which functions should not exist.
   c) all of the above.

2. The _____ in a test plan lets us know when to stop testing:
   a) test strategy
   b) test environment
   c) test completion criteria

3. It is important to carry out integration testing:
   a) so that you can make sure that unit testing has been carried out well.
   b) to check if the system can connect well with the other systems available on the network.
   c) to check how well the modules have been combined and how they perform.

4. _____ testing is performed by customers on their own site.
   a) Beta testing
   b) Alpha testing
   c) Black box testing

Activity 4.2.5

Let's say you want to test a flight reservation application. Develop two test scenarios with three test cases each.

Answers may vary

Test scenario	Test case	Test data	Expected results	Actual results	Pass/fail

Heartbeat Activity 4.1

Develop a Python code that displays the phases of the SDLC for different models. First, the user selects an SDLC model from a list of options. Then the program displays the relevant software development phases. After that, the program's menu is listed again.

```
loop = "3"
while (loop == "3"):
    print("---------------------------------")
    print("1. Waterfall")
    print("2. Agile")
    print("3. Exit Program")
    print("---------------------------------")
    ask = input("You want to know the phases of which SDLC model? ")
    if (ask == "1"):
        print ("-- Waterfall --")
        print("requirements analysis, defining specifications, designing, constructing, and testing
maintenance")


    elif(ask == "2"):
        print ("-- Agile --")
        print("Requirements gathering · Design the requirements · Construction/ iteration · Testing/
Quality assurance ·Deployment ·Feedback. ")


    elif(ask == "3"):
        loop = "X"
    else:
        print("Not an option, Try again")
print("End of Program")
```

LESSON PLAN

Grade: 11 A

Chapter 5: Operating Systems I
Section 1: Introduction to OS

1) Learning outcomes

Essential LOs

- ☐ Illustrate the purpose of an operating system.
- ☐ Compare the different types of software.

Strategies/Activities

Distance learning

- ★ LMS discussion forms
- ★ Microsoft teams group meetings, chats and assignments
- ★ E-Surveys
- ★ E-Polls

2) Keywords

- operating System (OS)

3) Resources

- Computer
- Student Book
- Workbook
- LMS
- Microsoft teams
- Tinkercad
- Video tutorials

4) Prior knowledge

- Computer programming

5) Assessment

Assessment for learning

- ★ Observations
- ★ Conversations
- ★ Notes
- ★ Work sample
- ★ Checklist
- ★ Diagnostics

Assessment as learning

- ★ Self-assessment
- ★ Peer-assessment
- ★ Presentation
- ★ Graphic Organizer
- ★ Collaboration
- ★ Homework

Assessment of learning

- ★ Workbook activities
- ★ Test
- ★ Quiz

6) Starter

Distance learning

- At the beginning of the lesson, introduce students to the lesson aim.
- Teacher to ensure all learning outcomes are outlined on LMS under the correct lesson.
- Use Microsoft teams screen sharing to show all learning outcomes and ensure understanding.
- Introduce the lesson's keywords through an LMS quiz or through an online game-based learning platform such as Kahoot or Quizlet.

7) Differentiation strategies

Differentiation to be met through designing various levels of activities. Some examples are outlined below:

Distance learning

- ★ LMS quizzes with visual aids MCQ, true or false and multi – response questions.
- ★ Differentiated LMS in class activities assigned to specific students.
- ★ Reflection through LMS surveys or polls.

Identify possible differentiation needs of the class based on prior classes and the starter activities.

8) Lesson activities

Distance learning

Teacher led activities	Student led activities
☐ Teacher to start an online lesson on Microsoft teams. ☐ Teacher to upload reading materials to LMS prior to the lesson. ☐ Students can view the books online via Al Diwan platform. ☐ Teacher to lead the class discussion on what is meant by operating systems and ask students to give examples on that topic. ☐ Teacher to introduce the history of operating systems and explain the operating systems in a computer structure. ☐ Teacher to introduce the operating systems managers ☐ Teacher to lead a class discussion on the different types of operating systems. ☐ Teachers to make sure that students complete activities 5.1.1 and 5.1.2. ☐ LMS activity	☐ Students to log on to Microsoft teams lesson ☐ Download any required material for the lesson from LMS. ☐ Open the required book on Al Diwan. ☐ LMS activity Students to complete activities 5.1.1 and 5.1.2.
Workbook activities ★ **5.1.1 – 5.1.2**	**9) Cross-curricular links** ▪ (Teacher to make cross-curricular links and activities where possible.)

10) Plenary

- Teacher to facilitate as students evaluate learning.
- Question pupils on what they have learned. Have learning outcomes been met? Has the lesson aim been achieved?
- Teacher to use LMS quiz or an online game-based learning platform such as Kahoot or Quizlet to check students understanding of the lesson objectives.
- Teacher to post homework activities through LMS and to assign a time frame for that for all unfinished activities.
- Teacher to create an LMS/Google/Microsoft surveys and share it with students where a checklist showing students' reflection will help self-reflect on the lesson for both teacher and students.
- Teacher to use LMS awards to give students recognition badges.

11) Reflection & Next steps

Activities that worked	Topics to be revisited

Answer Key – Workbook

Activity 5.1.1

Define an operating system in your own words.

Answers may vary

Activity 5.1.2

What are the main purposes of an operating system?

Answer:

- To provide an environment for a computer user to execute programs on computer hardware in a convenient and efficient manner
- To allocate the different resources of the computer as needed to solve the problem given. The allocation process should be as fair and efficient as possible.
- As a control program, it serves two major functions: (1) supervision of the execution of user programs to prevent errors and improper use of the computer, and (2) management of the operation and control of I/O devices.

Heartbeat Activity 5.1

Develop a simple quiz in Python that contains at least one multiple-choice question to test the user's knowledge of operating systems. For each item, the user gets two attempts to get the correct answer.

```
import random

# Dictionary of questions and answers

questions = {
        'Which of the following is not an operating system?':
        ('\na. Windows\nb. Linux\nc. Mac OS\nd. Oracle\n', 'd'),
        }

def ask_question(questions):
   '''Asks random question from 'questions 'dictionary and returns
      players's attempt and correct answer.'''

   item = random.choice(list(questions.items()))
   question = item[0]
   (variants, answer) = item[1]
   print(question, variants)
   attempt = input('\nHit \'a\', \'b\', \'c\' or \'d\' for your answer\n')
   return (attempt, answer)

# Questions loop
tries = 0
for questions_number in range(5):
      attempt, answer = ask_question(questions)
      if attempt not in {'a', 'b', 'c', 'd'}:
         print('INVALID INPUT!!! Only hit \'y\' or \'n\' for your response')
      elif attempt == answer:
         print('Correct')
         stop_asking = False
         break
      elif tries == 1: # Specify the number of tries to fail the answer
         print('Incorrect!!! You ran out of your attempts')
         stop_asking = True
         break
      else:
         tries += 1
         print('Incorrect!!! Try again.')
```

LESSON PLAN

Grade: 11 A

Chapter 6: Internet of Things I
Section 1: Basic Principles of IoT

1) Learning outcomes

Essential LOs

- ☐ Demonstrate the basic principles of IoT.
- ☐ Examine the main structure and components of an IoT system.
- ☐ Use an IoT platform to manipulate a controller board.

Strategies/Activities

Distance learning

- ★ LMS discussion forms
- ★ Microsoft teams group meetings, chats and assignments
- ★ E-Surveys
- ★ E-Polls

2) Keywords

- microprocessor
- Raspberry Pi
- Internet of Things (IoT)
- manipulate

3) Resources

- Computer
- Student Book
- Workbook
- LMS
- Microsoft teams
- Tinkercad
- Video tutorials

4) Prior knowledge

- Computer programming

5) Assessment

Assessment for learning

- ★ Observations
- ★ Conversations
- ★ Notes
- ★ Work sample
- ★ Checklist
- ★ Diagnostics

Assessment as learning

- ★ Self-assessment
- ★ Peer-assessment
- ★ Presentation
- ★ Graphic Organizer
- ★ Collaboration
- ★ Homework

Assessment of learning

- ★ Workbook activities
- ★ Test
- ★ Quiz

6) Starter

Distance learning

- At the beginning of the lesson, introduce students to the lesson aim.
- Teacher to ensure all learning outcomes are outlined on LMS under the correct lesson.
- Use Microsoft teams screen sharing to show all learning outcomes and ensure understanding.
- Introduce the lesson's keywords through an LMS quiz or through an online game-based learning platform such as Kahoot or Quizlet.

7) Differentiation strategies

Differentiation to be met through designing various levels of activities. Some examples are outlined below:

Distance learning

- ★ LMS quizzes with visual aids MCQ, true or false and multi – response questions.
- ★ Differentiated LMS in class activities assigned to specific students.
- ★ Reflection through LMS surveys or polls.

Identify possible differentiation needs of the class based on prior classes and the starter activities.

8) Lesson activities

Distance learning

Teacher led activities	Student led activities
☐ Teacher to start an online lesson on Microsoft teams. ☐ Teacher to upload reading materials to LMS prior to the lesson. ☐ Students can view the books online via Al Diwan platform. ☐ Teacher to lead the class discussion on what is meant by Internet and ask students to give their opinions on what is meant by 'things' in the term 'Internet of Things'. ☐ Teacher to provide examples on the applications of internet of things. ☐ Teacher to introduce the main sectors where IoT can be applied. ☐ Teacher to encourage students to think of examples of other sectors where IoT can be applied. ☐ Teachers to make sure that students complete activity 6.1.1. ☐ LMS activity ☐ Teacher to explain the structure of the IoT. ☐ Teacher to practically explain how to apply IoT through appendix and example 6.1.1 in the SB.	☐ Students to log on to Microsoft teams lesson. ☐ Download any required material for the lesson from LMS. ☐ Open the required book on Al Diwan. ☐ Students to complete activities 6.1.1 and 6.1.2.

Workbook activities	9) Cross-curricular links
★ **6.1.1 – 6.1.2**	▪ (Teacher to make cross-curricular links and activities where possible.)

10) Plenary

- Teacher to facilitate as students evaluate learning.
- Question pupils on what they have learned. Have learning outcomes been met? Has the lesson aim been achieved?
- Teacher to use LMS quiz or an online game-based learning platform such as Kahoot or Quizlet to check students understanding of the lesson objectives.
- Teacher to post homework activities through LMS and to assign a time frame for that for all unfinished activities.
- Teacher to create an LMS/Google/Microsoft surveys and share it with students where a checklist showing students' reflection will help self-reflect on the lesson for both teacher and students.
- Teacher to use LMS awards to give students recognition badges.

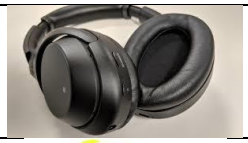
11) Reflection & Next steps

Activities that worked	Topics to be revisited

Answer Key – Workbook

Activity 6.1.1

Based on your understanding of what you have studied earlier, sort the following systems into **'IoT' or 'Not IoT'**.

System's description	Picture	Answer
Wireless Bluetooth headset		**Not IoT**
Wired remote-controlled car		**Not IoT**
Air conditioners that can be controlled from anywhere in the world		**IoT**
Wireless computer mouse		**Not IoT**
Smart home systems		**IoT**

Activity 6.1.2

Based on what you have studied earlier, give at least one example of an IoT application for the following domains.

Domain	Answer
Agriculture	**Greenhouse control systems** or **Soil and air moisture monitoring** or any other valid answer
Homes	**Smart lights** or **temperature and humidity monitoring** or any valid answer
Health	**Blood pressure and heart rate monitoring** or **Fitness monitoring systems** or any valid answer
Cities	**Smart traffic monitoring** or **Smart lighting that helps to save energy** or any valid answer

LESSON PLAN

Grade: 11 A

Chapter 6: Internet of Things I
Section 2: Control system inputs
Section 3: Using Microprocessor Boards

1) Learning outcomes

Essential LOs

- ☐ Construct a system using digital and analogue sensors.
- ☐ Develop a computer program to control various electronic components.
- ☐ Explain the function of sensors in electronic circuitry.
- ☐ Recommend various electronic sensors for suitable applications.
- ☐ Control a microprocessor-based system.
- ☐ Evaluate and modify codes to best solve a problem.

Strategies/Activities

Distance learning

- ★ LMS discussion forms
- ★ Microsoft teams group meetings, chats and assignments
- ★ E-Surveys
- ★ E-Polls

2) Keywords

- analogue sensor
- digital sensor
- library

3) Resources

- Computer
- Student Book
- Workbook
- LMS
- Microsoft teams
- Tinkercad
- Video tutorials

4) Prior knowledge

- Computer programming

5) Assessment

Assessment for learning

- ★ Observations
- ★ Conversations
- ★ Notes
- ★ Work sample
- ★ Checklist
- ★ Diagnostics

Assessment as learning

- ★ Self-assessment
- ★ Peer-assessment
- ★ Presentation
- ★ Graphic Organizer
- ★ Collaboration
- ★ Homework

Assessment of learning

- ★ Workbook activities
- ★ Test
- ★ Quiz

6) Starter

Distance learning

- At the beginning of the lesson, introduce students to the lesson aim.
- Teacher to ensure all learning outcomes are outlined on LMS under the correct lesson.
- Use Microsoft teams screen sharing to show all learning outcomes and ensure understanding.
- Introduce the lesson's keywords through an LMS quiz or through an online game-based learning platform such as Kahoot or Quizlet.

7) Differentiation strategies

Differentiation to be met through designing various levels of activities. Some examples are outlined below:

Distance learning

- ★ LMS quizzes with visual aids MCQ, true or false and multi – response questions.
- ★ Differentiated LMS in class activities assigned to specific students.
- ★ Reflection through LMS surveys or polls.

Identify possible differentiation needs of the class based on prior classes and the starter activities.

8) Lesson activities

Distance learning

Teacher led activities	Student led activities
☐ Teacher to start an online lesson on Microsoft teams. ☐ Teacher to upload reading materials to LMS prior to the lesson. ☐ Students can view the books online via Al Diwan platform.	☐ Students to log on to Microsoft teams lesson. ☐ Download any required material for the lesson from LMS. ☐ Open the required book on Al Diwan.
➢ **Section 2**	➢ **Section 2**
☐ Teacher to lead the class discussion on what is meant by sensors and ask students to give examples on sensors that they use in their everyday life. ☐ Teacher to give a brief introduction on analogue sensors by explaining the term 'analogue' and differentiating it from the term 'digital'. ☐ Teachers to make sure that students complete activity 6.2.1. ☐ LMS activities ☐ Teacher to explain the mentioned examples of analogue sensors and further give examples on where these sensors can be found. ☐ Teacher to explain the role of the raspberry pi in the system. ☐ Teacher to walk students through wiring, programming and running the code for analogue sensors on the raspberry pi. ☐ Teachers to make sure that students complete activities 6.2.2 – 6.2.3. ☐ LMS activities ☐ Teacher to give a brief introduction on digital sensors by explaining the term 'digital' and differentiating it from the term 'analogue'. ☐ Teacher to explain the mentioned examples of digital sensors and further give examples on where these sensors can be found. ☐ Teacher to explain the role of the raspberry pi in the system. ☐ Teacher to walk students through wiring, programming and running the code for digital sensors on the raspberry pi. ☐ Teachers to make sure that students complete activities 6.2.5 – 6.2.6. ☐ LMS activities	☐ LMS activities Students to complete activity 6.2.1. ☐ LMS activities Students to complete activates 6.2.2 and 6.2.3. ☐ LMS activities Students to complete activity 6.2.4. ☐ LMS activities Students to complete activities 6.2.5 and 6.2.6.

➢ **Section 3** ☐ Teacher to explain the LED blinking code where every part of the code must be explained clearly. ☐ Teacher to explain the PIR sensor code. ☐ Teacher to point out the similarities between the two codes; where they use the same libraries and end with the same lines of code. ☐ Teachers to make sure that students complete activities 6.3.1 – 6.3.2. ☐ LMS activities ☐ Teacher to lead the class discussion on how LDRs work and ask students to give examples on where this sensor can be used. ☐ Teacher to revise the meaning of analogue sensors in order to explain the LDR code. ☐ Teacher to remind students with the similarities between the codes. ☐ Teachers to make sure that students complete activities 6.3.3 – 6.3.4. ☐ LMS activities	➢ **Section 3** ☐ LMS activities Students to complete activities 6.3.1 and 6.3.2. ☐ LMS activities Students to complete activities 6.3.3 and 6.3.4.
➢ **Section 2** **Workbook activities (essential)** ★ **6.2.1, 6.2.2, 6.2.4, 6.2.5** **Workbook activities (non-essential)** ★ **6.2.3, 6.2.6** ➢ **Section 3** **Workbook activities (essential)** ★ **6.3.1, 6.3.2** **Workbook activities (non-essential)** ★ **6.3.3, 6.3.4**	**9) Cross-curricular links**  ▪ (Teacher to make cross-curricular links and activities where possible.)
10) Plenary ▪ Teacher to facilitate as students evaluate learning. ▪ Question pupils on what they have learned. Have learning outcomes been met? Has the lesson aim been achieved? ▪ Teacher to use LMS quiz or an online game-based learning platform such as Kahoot or Quizlet to check students understanding of the lesson objectives. ▪ Teacher to post homework activities through LMS and to assign a time frame for that for all unfinished activities. ▪ Teacher to create an LMS/Google/Microsoft surveys and share it with students where a checklist showing students' reflection will help self-reflect on the lesson for both teacher and students. ▪ Teacher to use LMS awards to give students recognition badges.	
11) Reflection & Next steps	
Activities that worked	**Topics to be revisited**

Answer Key – Workbook

Activity 6.2.1

Give an example of an analogue sensor that can be used with Raspberry Pi.

- Light-dependent resistor 'LDR' (also known as a photoresistor)
- Analogue sound sensor
- Vibration sensors
- Electric current sensors
- Magnetic fields hall effect sensors

Activity 6.2.2

Connect a potentiometer to the Raspberry Pi and write the Python program to read its position.

Note: The potentiometer is not a sensor, it is an input device. However, its hardware connection and programming is similar to regular analogue sensors.

In the space below, paste a picture of the circuit connection and write the code for the system.

Code

```
from time import sleep
from pymata_aio.pymata3 import PyMata3
from pymata_aio.constants import Constants

board = PyMata3(arduino_wait =1)

pot = 0

board.set_pin_mode(pot,Constants.ANALOG)

try:
    while True:
        val = board.analog_read(pot)
        print(val)

except Keyboardinterrupt:
    board.shutdown()
    print("\n done")
```

Activity 6.2.3

Build a project that reads the sound intensity of an analogue sound sensor. In the space below, paste a picture of the circuit connection and write the code for the system.

Code

```
from time import sleep
from pymata_aio.pymata3 import PyMata3
from pymata_aio.constants import Constants

board = PyMata3(arduino_wait =1)

sensor = 0

board.set_pin_mode(sensor,Constants.ANALOG)

try:
    while True:
        val = board.analog_read(sensor)
        print(val)

except Keyboardinterrupt:
    board.shutdown()
    print("\n done")
```

Activity 6.2.4

Give an example of a useful digital sensor that can be used with Raspberry Pi.

- **IR sensors**
- **Ultrasonic sensors**
- **Flame sensors**
- **Passive IR motion sensors 'PIR'**

Activity 6.2.5

Connect an IR sensor to the Raspberry Pi and write the Python program to print the values read by the sensor.
In this system, the IR sensor is used to differentiate between white and black surfaces.

In the space below, paste a picture of the circuit connection and write the code for the system.

Code

```
from time import sleep
from pymata_aio.pymata3 import PyMata3
from pymata_aio.constants import Constants
board = PyMata3(arduino_wait =1)
IR = 8
board.set_pin_mode(IR,Constants.INPUT)
try:
    while True:
        val = board.digital_read(IR)
        print(val)
except Keyboardinterrupt:
    board.shutdown()
    print("\n done")
```

Activity 6.2.6

Connect pushbutton to the Raspberry Pi and write the Python program to print the read values.

Note: The pushbutton is not a sensor, it is an input device. However, its hardware connection and programming similar to a regular digital sensor.

In the space below, paste a picture of the circuit connection and write the code for the system.

Code

```
from time import sleep
from pymata_aio.pymata3 import PyMata3
from pymata_aio.constants import Constants
board = PyMata3(arduino_wait =1)
button = 8
board.set_pin_mode(button,Constants.INPUT)
try:
    while True:
        val = board.digital_read(button)
        print(val)
except Keyboardinterrupt:
    board.shutdown()
    print("\n done")
```

Activity 6.3.1

Build a project that blinks an LED when a pushbutton is pressed and turns the LED off when the pushbutton is released.

In the space below, paste a picture of the circuit connection and write the code for the system.

Code

```
from time import sleep
from pymata_aio.pymata3 import PyMata3
from pymata_aio.constants import Constants

board = PyMata3(arduino_wait =1)

LED = 9
button = 8

board.set_pin_mode(LED,Constants.OUTPUT)
board.set_pin_mode(button,Constants.INPUT)

try:
    while True:
        val = board.digital_read(button)
        if val == 1:
            board.digital_write(LED,1)
            sleep(0.5)
            board.digital_write(LED,0)
            sleep(0.5)
        else:
            board.digital_write(LED,0)

except Keyboardinterrupt:
    board.shutdown()
    print("\n done")
```

Activity 6.3.2

Build a project that triggers a buzzer (blinks) when motion is detected and turns the buzzer off when no motion is detected.

In the space below, paste a picture of the circuit connection and write the code for the system.

Code

```
from time import sleep
from pymata_aio.pymata3 import PyMata3
from pymata_aio.constants import Constants

board = PyMata3(arduino_wait =1)

buzzer = 9
sensor = 8

board.set_pin_mode(buzzer,Constants.OUTPUT)
board.set_pin_mode(sensor,Constants.INPUT)

try:
    while True:
        val = board.digital_read(sensor)
        if val == 1:
            board.digital_write(buzzer,1)
            sleep(0.5)
            board.digital_write(buzzer,0)
            sleep(0.5)
        else:
            board.digital_write(buzzer,0)

except Keyboardinterrupt:
    board.shutdown()
    print("\n done")
```

Activity 6.3.3

Build a project that triggers a buzzer (blinks) when a potentiometer is in a position that is more than 50% of its range. Otherwise, the buzzer should turn off.

In the space below, paste a picture of the circuit connection and write the code for the system.

Code

```
from time import sleep
from pymata_aio.pymata3 import PyMata3
from pymata_aio.constants import Constants

board = PyMata3(arduino_wait =1)

buzzer = 9
pot = 0

board.set_pin_mode(buzzer,Constants.OUTPUT)
board.set_pin_mode(pot,Constants.ANALOG)

try:
    while True:
        val = board.analog_read(pot)
        if val > 512:
            board.digital_write(buzzer,1)
            sleep(0.5)
            board.digital_write(buzzer,0)
            sleep(0.5)
        else:
            board.digital_write(buzzer,0)

except Keyboardinterrupt:
    board.shutdown()
    print("\n done")
```

Activity 6.3.4

As you are now aware, many applications and systems are built using the Raspberry Pi. Carry out a quick online research to identify three applications of Raspberry Pi. Then, list the required input/output devices for each application.

Answers may vary